OPIC
OFFICE DE LA PROPRIÉTÉ INTELLECTUELLE DU CANADA

CIPO
CANADIAN INTELLECTUAL PROPERTY OFFICE

(12) (19) (CA) **Demande-Application**

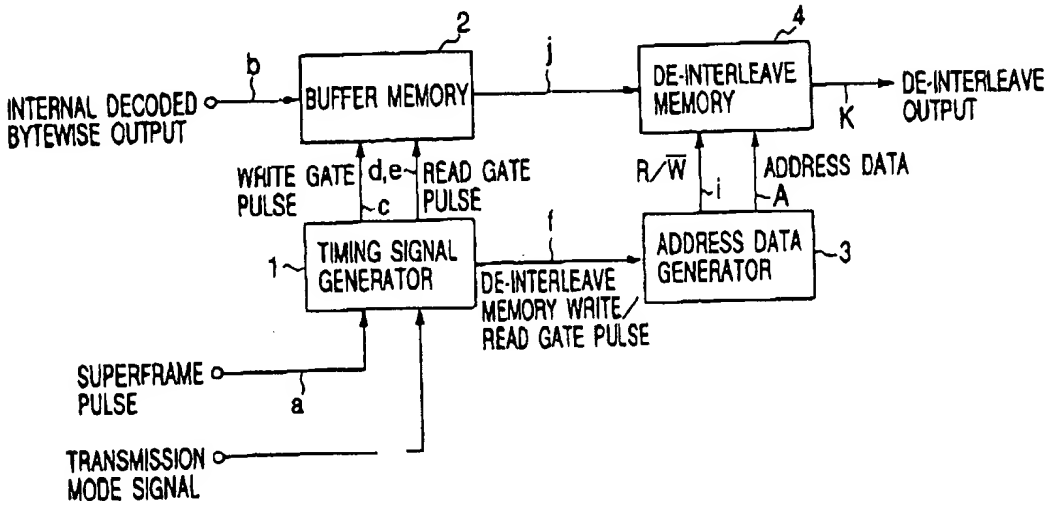
(21) (A1) **2,337,008**

(86) 1999/07/08
(87) 2000/01/27

(72) SHIRAISHI, KENICHI, JP
(72) SHINJO, SOICHI, JP
(72) HORII, AKIHIRO, JP
(71) KABUSHIKI KAISHA KENWOOD, JP

(51) Int.Cl.[6] H03M 13/22
(30) 1998/07/17 (10/218705) JP

(54) CIRCUIT DE DESENTRELACEMENT
(54) DE-INTERLEAVE CIRCUIT



(57) L'invention concerne un circuit de désentrelacement avec une mémoire inférieure. Un générateur de données adresses (3) fournit des données adresses (A) à une mémoire de désentrelacement (4) dans un ordre de désentrelacement. Chaque signal principal est lu depuis l'emplacement d'adresse situé dans une mémoire de désentrelacement (4) et spécifié par lesdites données adresses (A), le signal principal suivant étant désentrelacé puis écrit dans cet emplacement d'adresse de la mémoire. La mémoire de désentrelacement (4) susmentionnée ne nécessite donc que l'espace d'une supertrame.

(57) A de-interleave circuit is provided with less memory. An address data generator (3) supplies address data (A) to a de-interleave memory (4) in a de-interleave order. Each main signal is read from the address location in a de-interleave memory (4) specified by address data (A), and the following main signal is interleaved and written in that address location of the memory. As a result, the de-interleave memory (4) only requires space for one superframe.

DESCRIPTION

DE-INTERLEAVE CIRCUIT

Technical Field

The present invention relates to a de-interleave circuit used for a BS digital broadcasting receiver.

Background Art

As is already known, an interleave system in a BS digital broadcasting receiver performs block interleave, 8×203 bytes in terms of bytes, interleaving between slots with a same slot number in each frame in the superframe direction.

Here, an MPEG2-TS bucket of a main signal of a BS digital broadcasting signal comprises 1 slot made up of 204 bytes, 203 bytes including parity 16 bytes for correction of external code errors plus 1 byte of a TMCC (Transmission and Multiplexing Configuration Control) signal to indicate a synchronization signal, modulation system and error correction system, etc., with 48 slots forming 1 frame and 8 frames forming 1 superframe.

De-interleaving data interleaved as shown above requires a memory with a storage capacity for 2 superframes. This requires a memory with a storage capacity of a maximum of 155904 bytes (= 203 (bytes)×48 (slots)×8 frames×2 (superframes)).

However, configuring a conventional de-interleave circuit using a memory with a storage capacity amounting to 2 superframes

involves such a problem that the number of gates increases when the de-interleave circuit is implemented with an integrated circuit and at the same time the chip area increases.

It is an object of the present invention to provide a de-interleave circuit that requires less memory.

Disclosure of the Invention

The de-interleave circuit according to the present invention is a de-interleave circuit in a BS digital broadcasting receiver provided with address data generating means that generates address data A, in which assuming that the number of main signals in 1 slot is m, the depth of de-interleave is n, a specific address number assigned to a de-interleave memory is y, address data that specifies a data read/write address location is A, a modulo b is a remainder of a-αb (α: a natural number including 0),

when (y≠n×m-1), A= y×n to the xth power modulo (n×m-1), and

when (y=n×m-1), A=y,

address set number x, which is the number of times data of up to address number y is repeatedly specified, is such an x that when y=1, A=1 in A= y×n to the xth power modulo (n×m-1) and

when the value of y×n to the xth power is less than (n×m-1), A=(y×n to the xth power), and is characterized in that the main signal stored in the address location in a memory specified by the address data generated by the address generating means is read and the

following main signal, which is interleaved and input, is written in that address location.

In the de-interleave circuit according to the present invention, the main signal stored in the address location in memory specified by address data A generated by the address generating means is read. Since the following main signal, which is interleaved and input, is written in the address location, which has been substantially emptied by the aforementioned read, de-interleave can be performed with a storage capacity for a de-interleave memory and the de-interleave memory requires a storage capacity only 1/2 of the conventionally required storage capacity of 2 superframes, which means that the required area is reduced when the circuit is implemented with an integrated circuit.

In the de-interleave circuit according to the present invention, the address data generating means is provided with:

a base-m counter that counts the number of main signals input;

a base-S counter that counts a carry of the base m counter;

a multiplier that multiplies the count value of the base-S counter by (n×m);

offset value counting means that counts the number of main signals in the slot direction, shifts by one stage in the n direction when the main signal count value reaches (m-1) and counts main signals in the slot direction, and continues to count by repeating this procedure until the count value reaches (n×m); and

adding means that adds up the output of the multiplier and the count value of the offset value counting means, and is characterized in that the output of the adding means is deemed as address data A.

De-interleave is performed by reading a main signal from the address location in a memory specified by the address data output from the address generating means and writing the main signal.

Brief Description of the Drawings

Figure 1 is a block diagram showing a configuration of a de-interleave circuit according to an embodiment of the present invention;

Figure 2 is a schematic diagram to explain input/output data of a buffer memory in the de-interleave circuit according to the embodiment of the present invention;

Figure 3 is a schematic diagram to explain a timing signal of a timing signal generator in the de-interleave circuit according to the embodiment of the present invention;

Figure 4 is a schematic diagram to explain interleave, which is de-interleaved by the de-interleave circuit according to the embodiment of the present invention;

Figure 5 is a schematic diagram to explain the principle of de-interleave in the de-interleave circuit according to the embodiment of the present invention;

Figure 6 is a schematic diagram to explain the principle of de-interleave in the de-interleave circuit according to the embodiment of the present invention;

Figure 7 is a schematic diagram of an address space to explain de-interleave in the de-interleave circuit according to the embodiment of the present invention;

Figure 8 is a block diagram showing a configuration of an address data generator in the de-interleave circuit according to the embodiment of the present invention;

Figure 9 is a flow chart to explain address data generation in the de-interleave circuit according to the embodiment of the present invention;

Figure 10 is a flow chart to explain address data generation in the de-interleave circuit according to the embodiment of the present invention;

Figure 11 is a schematic diagram showing some addresses generated in the de-interleave circuit according to the embodiment of the present invention; and

Figure 12 is a schematic diagram showing some addresses generated in the de-interleave circuit according to the embodiment of the present invention.

Best Mode for Carrying Out the Invention

Hereinafter, the de-interleave circuit of the present invention will be explained according to an embodiment.

Figure 1 is a block diagram showing a configuration of the de-interleave circuit according to an embodiment of the present invention. The de-interleave circuit according to the embodiment of the present invention describes a case with a trellis coding system (hereinafter the trellis coding system will be referred to as "TC") 8PSK (coding rate r=2/3) of 46 slots and a convolutional coding and punctured coding system QPSK (coding rate r=1/2) of 1 slot.

Figure 3a shows a superframe pulse "a" output in synchronization with each superframe. An output (bytewise), which is a BS digital broadcasting signal received and then decoded, that is, a trellis code or internal decoded data "b" shown in Figure 3b, which is bytewise decoded by a trellis/Viterbi decoder, which is a decoder for a convolutional code are output in synchronization with superframe pulse "a." The internal decoded data "b" is written in a buffer memory 2 after receiving a write gate pulse "c" shown in Figure 3c output from a timing signal generator 1, which receives a superframe timing signal and transmission mode signal.

This write is stored when a main signal of 203 bytes is written in the buffer memory 2 made up of, for example, an FIFO, by a write gate pulse "c" and stored. A sync signal, a TMCC signal and a burst symbol signal are separated and are not written in the buffer memory 2 and no data is written during this period. The description of 4 bytes in Figure 3c shows a burst symbol period.

Here, the data rate of the bytewise data depends on the modulation system and coding rate, and if, for example, the data rate

of a TC8PSK code (coding rate r=2/3 (hereinafter the coding rate r=2/3 may be omitted)) is 1, then the data rate of a QPSK code (coding rate r=1/2 (hereinafter the coding rate r=1/2 may be omitted)) is 1/2. Therefore, this embodiment shows a case with the 8PSK code of 46 slots and QPSK code of 1 slot. The information bit transmission efficiency of the QPSK code is 1/2 of that of the 8PSK code and has 2 transmission slots because the signal is demodulated at a fixed symbol rate and 1 frame comprises 48 slots.

Moreover, the data rate of a BPSK code (coding rate r=1/2 (hereinafter the coding rate r=1/2 may be omitted)) is 1/4. During a period of a sync signal and the TMCC signal transferred with BPSK (coding rate r=1/2), the data rate is 1/4 of that of TC8PSK and 4 bytes of a burst symbol signal are transmitted with QPSK (coding rate r=1/2) and the data rate of the burst symbol signal is 1/2.

When a main signal of 203 bytes is written in the buffer memory 2 upon reception of the write gate pulse "c" output from the timing signal generator 1, the TMCC signal and the burst symbol signal are separated from the main signal and are not written in the buffer memory 2 as described above.

Upon reception of a read gate pulse "e" shown in Figure 3e output from the timing signal generator 1 starting at the time of generation of the read start pulse "d" shown in Figure 3d with a predetermined time difference from the superframe pulse shown in Figure 3a, reading of the internal decoded output "b" written in the buffer memory 2 is substantially stopped during the TMCC signal

and a burst symbol signal and a main signal "j" of 203 bytes (hereinafter, the description of "bytes" of 203 bytes may be omitted) is read in a burst-like manner from the buffer memory 2 at a certain speed. Furthermore, a de-interleave memory write/read gate pulse "f" shown in Figure 3f (a partially enlarged view is shown in Figure 3g) is output from the timing signal generator 1 to the address data generator 3.

From the address data generator 3 that has received the de-interleave memory write/read gate pulse "f," address data A shown in Figure 3h and R/W signal "i" shown in Figure 3i are output to the de-interleave memory 4, main signal "j" of 203 bytes shown in Figure 3j is written in the de-interleave memory 4 sequentially, and based on read address specification, de-interleave is performed when a read is performed from the de-interleave memory 4 and de-interleaved main signal "k" of 203 bytes shown in Figure 3k is output from the de-interleave memory 4.

For the R/W signal "i," a read instruction is executed at a high potential and a write instruction is executed at a low potential and the next supplied main signal is written at the address at which the main signal "i" is read by address specification according to address data A.

Then, detailed explanation will be given based on Figure 2. Figure 2 shows a data frame configuration of each processing process according to this embodiment.

Figure 2a shows internal decoded data "b" supplied to the buffer memory 2. The internal decoded data "b" shown in Figure 2a shows a frame configuration corresponding to 1 superframe when transmission is carried out 46 slots with a TC8PSK signal and 1 slot with a QPSK signal (coding rate r=1/2). 1 frame is configured by a sync/TMCC signal of 12 bytes followed by a TC8PSK main signal of 203 bytes, an empty section corresponding to a burst symbol signal of 4 bytes, a main signal of 203 bytes, an empty section corresponding to a burst symbol signal of 4 bytes,..., a "QPSK (coding rate r=1/2) main signal of 203 bytes and an empty section corresponding to a burst symbol signal of 4 bytes. Furthermore, 8 frames, the 0th frame to the 7th frame, make up 1 superframe.

The data stream of the internal decoded data "b" shown in Figure 2a is written and stored in the buffer memory 2 by the write gate pulse "c" only during a high-potential period of the gate pulse. As shown in Figure 2b, the main signals, that is, signals other than the sync signal, TMCC signal and burst symbol signal section are written in the buffer memory 2.

Upon reception of superframe pulse "a," the timing signal generator 1 generates read start signal "d," which is delayed by a certain time, and at this timing, read gate pulse "e" is output. Upon reception of read gate pulse "e," a read is performed from the buffer memory 2 at a certain rate. Suppose the reading rate is the rate of 8PSK, which is the basic rate.

Read gate pulse "e" comprises a 48-byte stop of sync/TMCC signal section, 203-byte read, 4-byte stop, 203-byte read, 4-byte stop, ..., 203-byte stop and 4-byte stop. The reason that the sync/TMCC signal section has 48 bytes is that the data rate is 1/4 during a write, while the data rate is quadrupled (12 bytes×4=48) during a read.

The reason that the pulse ends with a 203-byte stop is that the information bit transmission efficiency of the main signal QPSK code is 1/2 of that of the TS8PSK code and has 2 transmission slots because it has been demodulated at a fixed symbol rate, but it has been changed to 203 bytes by internal decoding and dummy data is inserted into this section. Figure 2c shows the dummy data inserted in the section assigned to the 203-byte stop.

Here, it is assumed that no burst symbol signal is written in the buffer memory 2, but it is also possible to write the burst signal together with the main signal and also read the burst signal. In this case, the 4-byte low potential section of the buffer memory write gate pulse "c" shown in Figure 3c and the read gate pulse "e" shown in Figure 3e can be changed to a high potential section.

From the timing signal generator 1, the de-interleave memory write/read gate pulse "f" shown in Figure 3f is output to the address data generator 3. Upon reception of the de-interleave memory write/read gate pulse "f," address data A and R/W signal "i" are output from the address data generator 3 to the de-interleave memory 4 and de-interleave is performed.

Output data "j" including the dummy data shown in Figure 2c from the buffer memory 2 is written and read to/from the de-interleave memory 4 through address data A and R/W signal "i" whose timing is matched with the de-interleave memory write/read gate pulse. Figure 3g to 3k show enlarged views of the de-interleave memory write/read gate pulse 6 for 1 slot and at the same time show enlarged views of address data A and R/W signal "i" for 1 slot and write data "i" and read data "k" in the de-interleave memory.

As is also apparent from Figure 8, de-interleaved data "k" preceding by 1 superframe is read first and then data "j" to be currently de-interleaved is written. Furthermore, by writing write data "j" at the same address that data "k" is read, the timings for write data and read data become the same. The read data "k" is de-interleaved data and has the same frame configuration as that shown in Figure 2c.

De-interleave is performed by an operation to write or read to/from the de-interleave memory 4. Then, the address data generator 3 will be explained.

Before explaining the address data generator 3 in detail, the principle of address generation will be explained based on Figure 4, Figure 5 and Figure 6.

In Figure 4, Figure 5 and Figure 6, "m" denotes a basic unit (= 203 bytes in the case of BS digital broadcasting) of data length and "n" denotes a depth of interleave (= 8 in the case of BS digital broadcasting (the number of frames that make up 1 superframe)),

and supposing m=5 and n=4 here for simplicity, Figure 4, Figure 5 and Figure 6 show an example of de-interleave of n×m=4×5.

The upper row of the de-interleave matrix in Figure 4a and Figure 5a, b and c indicates address data and the lower row indicates data entered. As shown in Figure 4a, data stream D[0], D[1], D[2], D[3], ..., D[18], D[19] are written for address data A[0], A[1], A[2], A[3], ..., A[18] and A[19]. This data is sequentially written in the horizontal direction as shown in Figure 4c, and interleaved by being read sequentially in the vertical direction as shown in Figure 4b.

This state is shown in Figure 5a and this state is assumed to be the case where address set number x=0. In this state, address data A[0], A[1], A[2], A[3], A[4], A[5], ..., A[18] and A[19] are sequentially output from the address data generator 3, and interleaved data stream D[0], D[5], D[10], D[15], D[1], D[6] ..., D[14], D[19] of a depth of 4 for data stream D[0], D[1], D[2], D[3], D[4], D[5], ..., D[18], D[19] are written.

In this case, the address data output from the address data generator 3 is of a simple increment and this is called "basic address set." For the read address data from the de-interleave memory 4, address data of A[0], A[4], A[8], A[12], A[16], A[1], ..., A[15], A[19] are output sequentially from the address data generator 3 so that the data stream after being read becomes the state before being interleaved D[0], D[1], D[2], D[3], D[4], D[5], ..., D[18], D[19]. This address set number x is assumed to be 1.

The next data stream is written at the address at which the read is performed. This data stream is D'[0], D'[5], D'[10], D'[15], D'[1], D'[6], ..., D'[14], D'[19] and Figure 5b shows the state in which this data stream is written.

Likewise, in the case where data is de-interleaved, if addressing is performed so that the address data is output as A[0], A[16], A[13], A[10], A[7], A[4], ..., A[3], A[19], then data D'[0], D'[1], D'[2], D'[3], D'[4], D'[5], ..., D'[18], D'[19] is obtained. This is the state of address set number x=2.

Moreover, in the order of this address data, that is, in the order of address locations specified by address data in which data is read, data D"[0], D"[5], D"[10], D"[15], D"[1], D"[6], ..., D"[14], D"[19] is written (Figure 5c) and if data is read from the positions specified by address data A[0], A[7], A[14], A[2], A[9], A[16],..., A[12], A[19] of address set number x=3, then data D"[0], D"[1], D"[2], D"[3], D"[4], D"[5],... ..., D"[18], D"[19] is output.

Figure 6 shows the progression of address data A corresponding to address number y corresponding to specific address data assigned to the de-interleave memory 4 and address set number x, which is the number of times data of up to address number y, which corresponds to the number of times data is read/written, is repeatedly specified, and if the case with address set number x=2 is explained as an example, it is shown that specific address numbers A[0], A[16], A[13], A[10], A[7], A[4], ..., A[3], A[19] which are assigned

to the de-interleave memory 4 are output as address data from the address data generator 3.

Here, the de-interleave procedure can be summarized as follows:

a. Read with address set number x=0 (omitted at first)

b. Write with address set number x=0

c. Read with address set number x=1

d. Write with address set number x=1

e. Read with address set number x=2

f. Write with address set number x=2

.

.

.

q. Read with address set number x=8

r. Write with address set number x=8

s. Read with address set number x=9 (=0)

and thus data, which is input, is written at the address of the address data whose data has been read. In this way, it is seen that specification of address data with de-interleave of n×m=4×5 circulates in a cycle of address set number x of 9. The cycle of address set number x is 9 and this is described as cycle X. Cycle X=0 has no meaning, and therefore X=0 is excluded.

Then, this addressing will be expressed in a general expression. Suppose the depth of de-interleave is n, the number of basic unit data items is m, the address set number is x, the address number is

y and the address data is A, then address data A is specified as follows:

A=y×n to xth power modulo (n×m-1)

(y≠n×m-1) ... (1)

A=y (y=n×m-1) ... (2)

In this case, suppose a modulo b is a remainder of a-αb (α is a natural number including 0) and if a is smaller than b, a modulo b is a.

Cycle X of address set number x is obtained by obtaining x, which satisfies y=1 and A=1 in expression (1) and in this case, X=9 as described above.

The examples shown in Figure 4, Figure 5 and Figure 6 are expressed in a general expression with respect to de-interleave on a two-dimensional address matrix, and are also applicable to de-interleave of a BS digital broadcasting signal. With the BS digital broadcasting signal, interleave with a depth of 8 is carried out between the same slots in the frame direction. A main signal of 203 bytes is assigned to 1 slot and it is possible to handle the main signal as a set of 48 (slot) two-dimensional matrices of 203 (bytes)×8 (frames). That is, a memory space corresponding to 1 superframe is divided into 48 portions and 203×8 de-interleaves are performed in each area.

In this embodiment, a 1-superframe memory matrix is defined as shown in Figure 7. Suppose m=203 (bytes)×n=8 (frames)×S=48 (slots) and the address increments from address 0 in the m direction

in a 203×8 two-dimensional matrix, then shifts by one stage in the n direction and increments in the m direction again,... and so on. Then, the final address of the 1 slot is 1623. Also suppose that an offset is added by n×m=1624 at a time in the direction from the 1st slot to the 48th slot.

In this case, too, cycle X of address set number x is obtained from x, which satisfies y=1 and A=1 in expression (1) and cycle X of address set number x is 180.

Figure 8 shows the embodiment of the address data generator 3 and Figure 9 and Figure 10 show a flow chart indicating the action thereof.

As shown in Figure 8, the address data generator 3 is provided with a strobe pulse generator 50, a slot number detection section 51 that specifies slot numbers in cooperation with the strobe pulse generator 50 and a modulo operation section 52 that performs modulo operations and sends address data in cooperation with the strobe pulse generator 50 and the slot number detection section 51. Here, m=203 (the number of bytes of a main signal in 1 slot), n=8 (depth of de-interleave), S=48 (the number of slots in 1 frame), F=8 (the number of frames making up 1 superframe and F=n=8) and X=180 (cycle of address set count x). Here, the modulo operation section 52 except an adder 70, which will be described later, corresponds to offset value counting means.

The strobe pulse generator 50 comprises a base-m counter 53 that, supplied with de-interleave memory write/read gate pulses

output from the timing signal generator 1, receives clock pulses output during the interval that the de-interleave memory write/read gate pulses are at a high potential and counts clock pulses, a base-S counter 54 that counts the carry output of the base-m counter 53, a base-F counter 55 that counts the carry output of the base-S counter 54, a base-X counter 56 that counts the carry output of the base-F counter 55, and a decoder 57 that receives count value mcnt of the base-m counter 53, count value Scnt of the base-S counter 54, count value Fcnt of the base-F counter 55 and count value xcnt of the base-X counter 56 and generates a strobe pulse.

The count value of the base-S counter 54 is incremented every time the base-m counter 53 counts clock pulses 0 to 203 times, which means that the base-S counter 54 detects slot numbers. The slot number detection section 51 comprises a multiplier 58 that receives the count value of the base-S counter 54 and multiplies the count value by (n×m) and generates the slot number start address data 0, 1624, 3248, ..., 76328 based on the count value of the base-S counter 54. Data A0, which will be described later, is added to this output from the slot number detection 51 and address data A is obtained.

The modulo operation section 52 is provided with a setter 59 that initializes the A offset register 60 to set value "1," an A offset register 60 that receives a strobe pulse "sa" and registers the register value of the R offset register 61, an R offset register 61 that receives strobe pulse "sb" and registers address data A0, an adder 62 that

adds up the register values of address data A0 and A offset register 60, a comparator 64 that compares the addition output of the adder 62 and the set value (n×m) of the setter 63, a subtractor 66 that based on the output of the comparator 64 when (addition output of the adder 62≥set value (n×m) of the setter 63), outputs a value obtained by subtracting the set value (n×m-1) of the setter 65 from the addition output of the adder 62 as address data A" and outputs the addition output of the adder 62 as address data A" based on the output of the comparator 64 when (addition output of the adder 62≥set value (n×m) of the setter 63) is not the case, an Amcnt register 67 that receives strobe pulse "sc" and registers the address data A" output from the subtractor 66 as a register value, a selector 68 that selects one of address data A" output from the subtractor 66 and the register value of the Amcnt register 67 through select pulse "sp," a latch 69 made up of a DF/F that delays address data A' output from the selector 68 by 1 clock pulse and an adder 70 that adds up latched address data A0 and the output of the multiplier 58, and designates the output of the adder 70 as address data A.

Strobe pulse "sa" to the A offset register 60 is output in synchronization with the carry output of the base-F counter 55. However, in the case of xcnt=X-1, "1" is registered and in the case of xcnt≠X-1, register value Rofset of the R offset register 61 is registered. Strobe pulse "sb" to the R offset register 61 is output when count value Fcnt of the base-F counter 53 is 0, count value Scnt of the base-S counter 52 is 0 and count value mcnt of the base-m counter

53 is n. Strobe pulse "sc" to the Amcnt register 67 is output when count value Scnt of the base-S counter 52 is 0 and count value mcnt of the base-m counter 53 is 0. Select pulse "sp" to the selector 68 is output when count value mcnt of the base-m counter 53 is m-1 and count value mScnt of the base S counter 52 is not S-1 and the register value of the Amcnt register 67 is selected.

The adder 62 adds up the value registered in the A offset register and address data A0 and the addition result is sent to the comparator 64 and subtractor 66. Address data A0 is address data in a 203×8 two-dimensional matrix, and the result of an addition of count value Scnt of the base-S counter 54 that counts the number of slots multiplied by n×m by the multiplier 58 (that is, an offset in the slot direction) to address data A0 becomes address data A.

The comparator 64 outputs a subtraction instruction to the subtractor 66 when the addition output of the adder 62 becomes (n×m(=1624) or greater and the subtractor 66 receives the subtraction instruction and subtracts (n×m-1) set in the setter 65 from the addition output from the adder 62. When the addition output of the adder 62 is not equal to or greater than (n×m(=1624), no subtraction is carried out and the addition output of the adder 62 is directly output from the subtractor 66.

In the Amcnt register 67, the address data A" at the time of the occurrence of strobe pulse "sc" is registered, and when the selector 68 receives select signal "sp," register value A" in the Amcnt register 67 is selected and output. The output from the selector 68 is deemed

as address data A'. The address data A' is latched by the latch 69 and the latch output is deemed as address data A0. Furthermore, the R offset register 61 receives strobe pulse "sa" and registers data A0 at that time. Moreover, the register value of the R offset register 61 is output to the A offset register 60 and registered in the A offset register 60 upon reception of strobe pulse "sa."

The base-m counter 53, base-S counter 54, base-F counter 55, base-X counter 56 and latch 69 operate on a common clock pulse, but stop operating when the de-interleave memory write/read gate pulse is at a low potential.

The action of the address data generator 3 will be explained based on the flow charts in Figure 9 and Figure 10.

When de-interleave is started, register value Aofset of the A offset register 60 is initialized to 1 and the count value xcnt of the base-X counter 56, that is, the address set number x is initialized to 0 (step S1). Furthermore, the count value mcnt of the base-m counter 53, count value Scnt of the base-S counter 54 and count value Fcnt of the base-F counter 55 are initialized to 0, and the latch 69 is also initialized and address data A0 is also initialized (step S2). At this time, data A" is registered in the Amcnt register 67, but in this case 0 is registered (step S3). Moreover, strobe pulse "sc" becomes a high potential when the count value mcnt of the base-m counter 53 is 0, the count value Scnt of the base-S counter 52 is 0, and therefore a value is registered in the Amcnt register 67 for every increment of the base-F counter 55.

The register value to the Amcnt register 67 output via the selector 68 is latched by the latch 69 and address data A0 is confirmed (step S4). In step S5 in which the decoder 57 checks whether count value Fcnt of the base-F counter 55 is 0 or not, count value Scnt of the base-S counter 54 is 0 or not, or count value mcnt of the base-m counter 53 is mcnt=n=8 or not (step S5), in the case where it is decided that count value Fcnt of the base-F counter 55 is 0, count value Scnt of the base-S counter 54 is 0 and count value mcnt of the base-m counter 53 is mcnt=n=8, strobe pulse "sb" is output, address data A0 is registered in the R offset register 61 (step S6) and step S7 is executed. However, since count value mcnt of the base-m counter 53 is 0, step S5 to step S7 are executed.

Until count value mcnt of the base-m counter 53 becomes m-1 (=202) in step S7, count value mcnt of the base-m counter 53 is incremented after step S7 though not shown in Figure 9, and then step S8 is executed. In step S8, the adder 62 adds up the address data A0 and the register value registered in the A offset register 60 (step S8). In the case where addition output A"' of the adder 62 is (n×m(=1624) or greater (step S9), (n×m-1(=1623) is subtracted from the addition output A"' and the process is repeated from step S4 (step S10). In the case where addition output A"' of the adder 62 is not equal to (n×m(=1624) or greater, the process is repeated from step S4 after step S9.

If the operation so far is checked against the aforementioned addressing general expressions, the register value of the A offset

register 60 is equal to n to the xth power (when n to the xth power exceeds (n×m-1), the register value of the A offset register 60 is equal to the remainder when (n×m-1) is subtracted repeatedly) and y×n to the xth power is equal to an accumulated sum of n to the xth power. Moreover, since data A"' never exceeds two times (n×m-1), in the case (n×m-1) is exceeded, the configuration of a modulo operation of (n×m-1) can be simplified by subtracting (n×m-1). At the final address at which data A"' becomes equal to (n×m-1), subtracting (n×m-1) results in 0, causing a problem.

However, since it is only at the final address that A"' becomes equal to (n×m-1), this problem can be avoided by changing the subtraction condition so that (n×m-1) is subtracted when (n×m) is exceeded. This is equivalent to simplifying the condition of A=y as a combination of y= n×m-1 in the above-described addressing general expressions.

In the case where the count value mcnt of the base-m counter 53 is m-1 (=202), the step of a conditional branch based on count value Scnt of the base-S counter 54 is executed (step S11). Step S12 is executed until count value Scnt of the base-S counter 54 becomes S-1 (=47) and the process is repeated from step S4 after step S12. In step S12, count value Scnt of the base-S counter 54 is incremented, count value mcnt of the base-m counter 53 is reset and the register value of the Amcnt register 67 is output as address data A' (step S12), that is, the selector 68 selects the register value of the Amcnt register

67. This operation is carried out to equalize the initial value of address data A of each slot in a frame.

For example, address data A0 of each slot begins with 0 on the 1st frame and begins with 203 on the 2nd frame. Therefore, on the 2nd frame, it is necessary to load 203 in address data A0 every time the slot is changed. Since count value Fcnt of the base-F counter 55 is currently 0, that is, the 1st frame, "0," which is registered in the Amcnt register 67, is loaded for every increment of the slot. The operation above is repeated until count value Scnt of the base-S counter 54 becomes S-1 (=47).

In the case where count value Scnt of the base-S counter 54 becomes S-1 (=47), a conditional branch step based on count value Fcnt of the base-F counter 55 is executed (step S13). In the case where the count value Fcnt of the base-F counter 55 is less than F-1 in step S13, step S14 is executed, count value Fcnt of the base-F counter 55 is incremented and count value Scnt of the base-S counter 54 and count value mcnt of the base-m counter 53 are reset (step S14). Then, A0 is added to the A offset register 60 (step S15). This is because the initial value of address data A0 when the frame is changed is a value next to data A0, which is the final value of the previous frame.

That is, when address set number x is 0, since final address data A0 on the first frame is 202 and at the beginning of the 2nd frame, the register value of the A offset register 60 is "1," 202 + 1 = 203. Furthermore, as a result of steps S15, step S16 is executed in

which data A'≥(n×m) is checked, and as a result of step S16, step S17 is executed selectively, then step S3 is executed. It is the same as the aforementioned case that (n×m-1) is subtracted when data A' exceeds (n×m) (step S17). Furthermore, this result is registered in the Amcnt register 67 in step S3 and becomes a value loaded every time the slot is changed.

In the case where count value Fcnt of the base-F counter 55 becomes F-1 (=7) in step S13, according to a conditional branch (step S20) based on count value xcnt of the base-X counter 56 whose de-interleave for 1 superframe is completed at this time, step S21 is executed in the case where address set number x does not reach x=X-1 (= 179) and the register value of the R offset register 61 registered in step S6 is registered in the A offset register 60 (step S21). Furthermore, address set number x is incremented (step S22).

This operation will be explained according to the general expressions of addressing; expression (1) and expression (2).

The register value of the A offset register 60 is equal to address data A (value of A=1×n to the xth power) (equal to a remainder when the subtraction is repeated with (n×m-1) in the case where the value of n to the xth power exceeds (n×m-1) as described above) and register value Aofset' of the A offset register 60 of the next address set becomes 1×n to the (x+1)th power = 1×n to the xth power×n = 8×n to the xth power, and this is equal to address data A when y=8. In other words, if address data A of current y=8 is stored, this becomes Aofset', which is the register value of the A offset register 60 of the

next address set number x and this means that a calculation circuit can be omitted. Since in step S5 and step S6, address data A when y=mcnt=n=8 is stored in the R offset register 61, this is written in the A offset register 60 before moving to the next address set. Moreover, when count value xcnt of the base-X counter 56 becomes X-1 (= 179), all values are initialized.

Figure 11 and Figure 12 show part of address data A created by this embodiment. Due to limitations of space, address set number x of up to 17 and address number y of up to 50 are shown.

As shown above, according to the de-interleave circuit according to this embodiment, reads are performed ahead of writes on address data A to the de-interleave memory 4 generated by the address data generator 3 as read (R), write (W), read (R), write (W), ... and writing data in an address which is emptied by data read improves the efficiency of memory utilization.

On the other hand, it is possible to provide, for example, two address data generators, use one for read address data generation only and the other for write address data generation only and thereby carry out reading at high speed in superframe units. In this case, the address data must be the same address data. For example, according to the timing shown in Figure 3, a pattern of a 4-byte stop period provided after both 203-byte read/write processing is repeated, but it is also possible to perform reads consecutively without any 4-byte stop period for reads only.

Industrial Applicability

As described above, the de-interleave circuit according to the present invention makes it possible to carry out de-interleave with a storage capacity for 1 superframe, which brings about an effect of reducing the number of parts and parts cost. Moreover, when implemented in an integrated circuit, the present invention can reduce the number of gates and drastically reduce the chip area compared to the prior art.

Revised Claims Sheet

1. (Amended) A de-interleave circuit for recovering an original data arrangement from an interleaved data arrangement of a main signal, said interleaved data arrangement being formed in such a way to execute interleaving between slots having an identical frame slot number in a superframe direction, said de-interleave circuit, comprising:

a de-interleave memory having a storage capacity of one superframe; and

address data generating means for generating address data to specify a data read/write address location to said de-interleave memory,

characterized in that said main signal stored in the memory address location specified by the address data generated by the address generating means is read and a following main signal to be interleaved and inputted is written in said address location.

2. The de-interleave circuit according to claim 1, characterized in that said address data generating means generates address data A, assuming that the number of main signals in 1 slot is m, the depth of de-interleave is n, a specific address number assigned to said de-interleave memory is y, address data that specifies a data read/write address location is A, and a modulo b is a remainder of a-αb (α: a natural number including 0),

when (y=n×m-1), A=y,

address set number x, which is the number of times data of up to address number y is repeatedly specified, is such an x that when y=1, A=1 in A= y×n to the xth power modulo (n×m-1) and

when the value of y×n to the xth power is less than (n×m-1), A=(y×n to the xth power).

3. The de-interleave circuit according to claim 2, characterized in that said address data generating means comprising:

a base-m counter that counts the number of main signals input;

a base-S counter that counts a carry of the base-m counter;

a multiplier that multiplies the count value of the base-S counter by (n×m);

offset value counting means that counts the number of main signals in the slot direction, shifts by one stage in the n direction when the main signal count value reaches (m-1) and counts main signals in the slot direction, and continues to count by repeating this procedure until the count value reaches (n×m); and

adding means that adds up the output of the multiplier and the count value of the offset value counting means,

in which the output of the adding means is deemed as address data A.

ABSTRACT

To provide a de-interleave circuit used for a BS digital broadcasting receiver. The de-interleave circuit is provided with less memory. An address data generator (3) supplies address data (A) to a de-interleave memory (4) in a de-interleave order. Each main signal is read from an address location in the de-interleave memory (4) specified by address data (A), and a following main signal is interleaved and written in that address location of the memory. As a result, the de-interleave memory (4) only requires space for one superframe.

FIG. 1

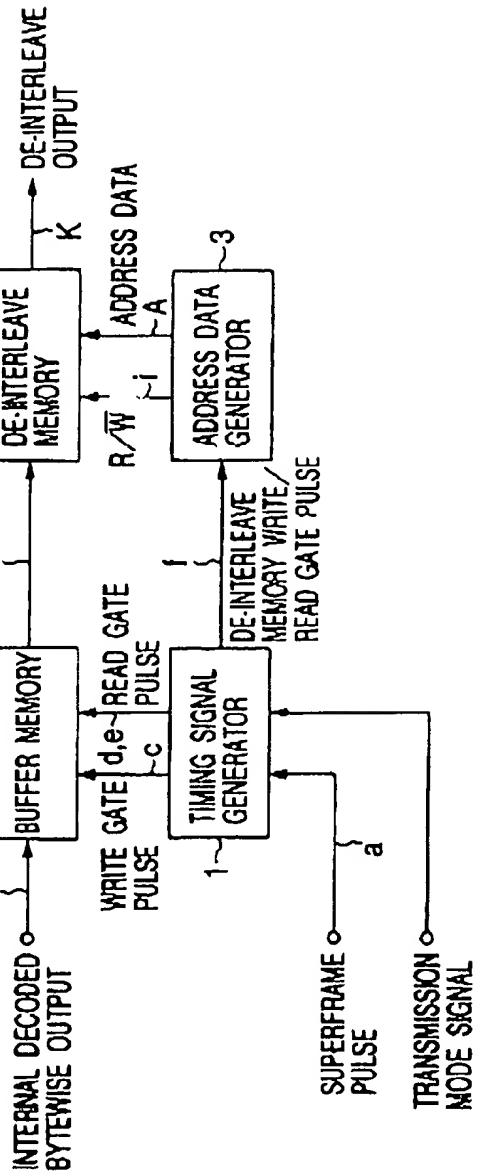
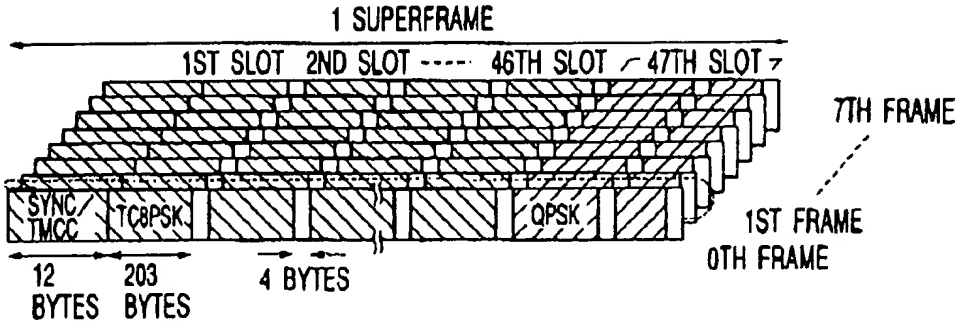
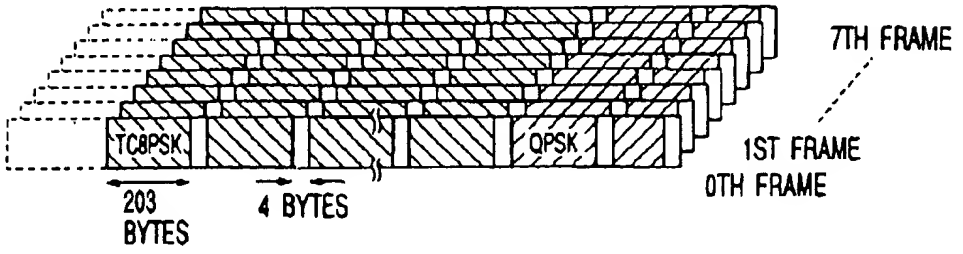

INTERNAL DECODED BYTEWISE OUTPUT
b
BUFFER MEMORY
2
WRITE GATE PULSE
d,e
READ GATE PULSE
c
TIMING SIGNAL GENERATOR
1
SUPERFRAME PULSE
a
TRANSMISSION MODE SIGNAL
DE-INTERLEAVE MEMORY WRITE/READ GATE PULSE
f
ADDRESS DATA GENERATOR
3
ADDRESS DATA
A
R/W
i
DE-INTERLEAVE MEMORY
4
DE-INTERLEAVE OUTPUT
K
j

FIG. 2

(a)


1 SUPERFRAME
1ST SLOT
2ND SLOT
46TH SLOT
47TH SLOT
7TH FRAME
1ST FRAME
0TH FRAME
SYNC/TMCC
TC8PSK
QPSK
12 BYTES
203 BYTES
4 BYTES

(b)


7TH FRAME
1ST FRAME
0TH FRAME
TC8PSK
QPSK
203 BYTES
4 BYTES

(c)

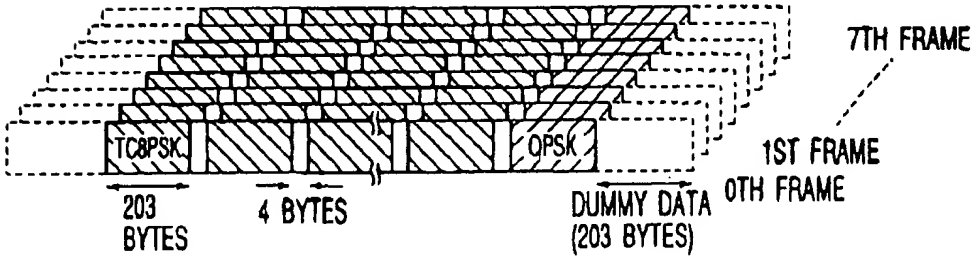

7TH FRAME
1ST FRAME
0TH FRAME
TC8PSK
QPSK
203 BYTES
4 BYTES
DUMMY DATA (203 BYTES)

FIG. 3

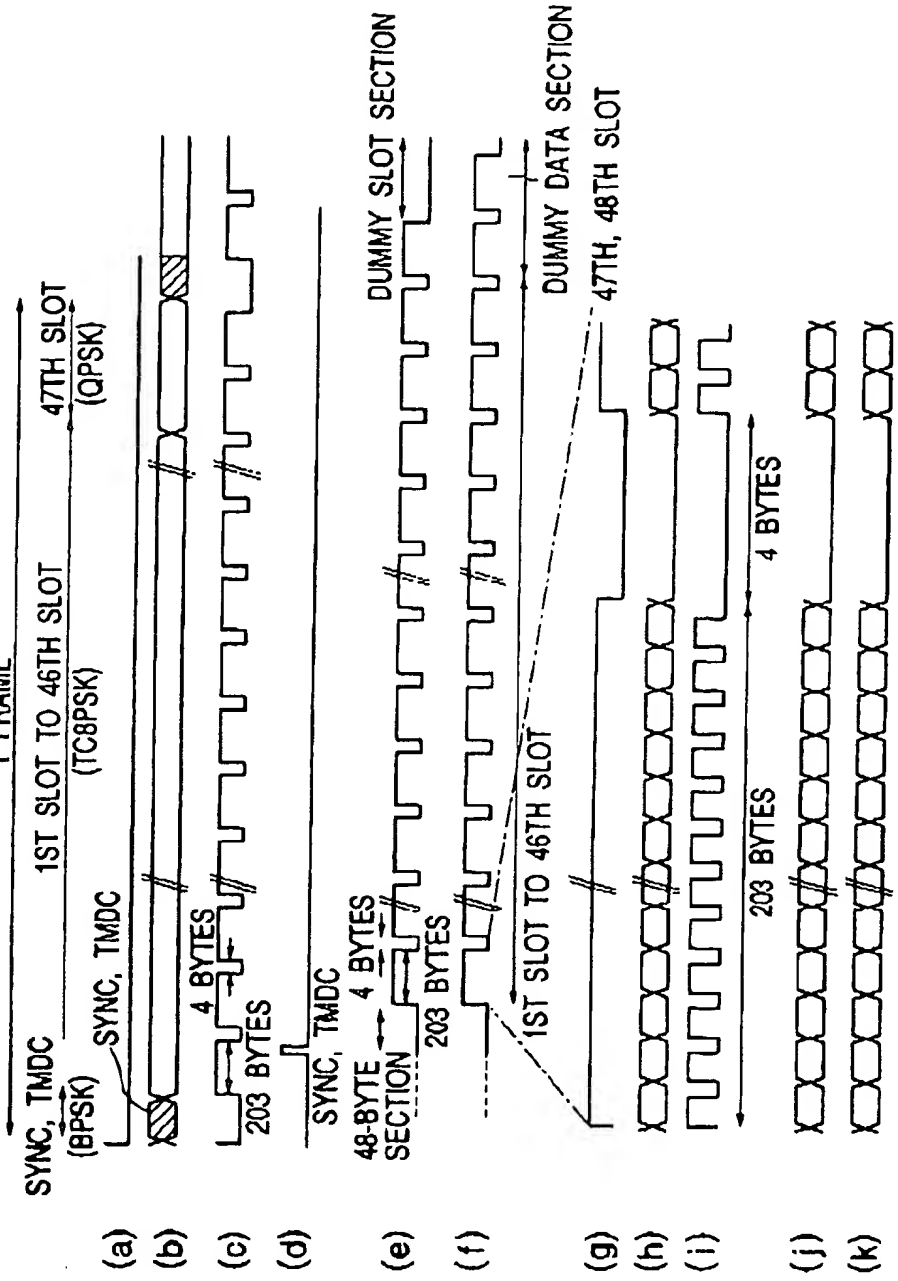
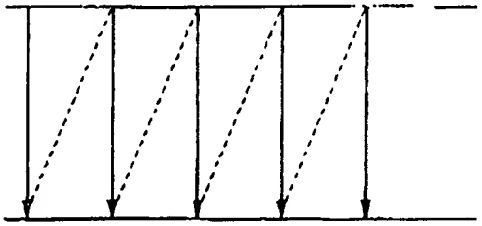

1 FRAME
SYNC, TMDC
(BPSK)
1ST SLOT TO 46TH SLOT
(TC8PSK)
47TH SLOT
(QPSK)
SYNC, TMDC
4 BYTES
203 BYTES
SYNC, TMDC
48-BYTE SECTION
4 BYTES
203 BYTES
DUMMY SLOT SECTION
1ST SLOT TO 46TH SLOT
DUMMY DATA SECTION
47TH, 48TH SLOT
203 BYTES
4 BYTES
(a)
(b)
(c)
(d)
(e)
(f)
(g)
(h)
(i)
(j)
(k)

FIG. 4

(a)

A[0] D[0]	*A[1]* D[1]	*A[2]* D[2]	*A[3]* D[3]	*A[4]* D[4]
A[5] D[5]	*A[6]* D[6]	*A[7]* D[7]	*A[8]* D[8]	*A[9]* D[9]
A[10] D[10]	*A[11]* D[11]	*A[12]* D[12]	*A[13]* D[13]	*A[14]* D[14]
A[15] D[15]	*A[16]* D[16]	*A[17]* D[17]	*A[18]* D[18]	*A[19]* D[19]

(b)



(c)

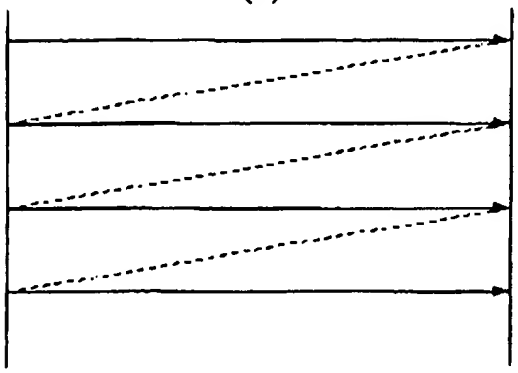


FIG. 5

(a)

m=5 (WRITE DIRECTION)

DEPTH n=4

A[0] D[0]	A[1] D[5]	A[2] D[10]	A[3] D[15]	A[4] D[1]
A[5] D[6]	A[6] D[11]	A[7] D[16]	A[8] D[2]	A[9] D[7]
A[10] D[12]	A[11] D[17]	A[12] D[3]	A[13] D[8]	A[14] D[13]
A[15] D[18]	A[16] D[4]	A[17] D[9]	A[18] D[14]	A[19] D[19]

(b)

A[0] D'[0]	A[1] D'[6]	A[2] D'[12]	A[3] D'[18]	A[4] D'[5]
A[5] D'[11]	A[6] D'[17]	A[7] D'[4]	A[8] D'[10]	A[9] D'[16]
A[10] D'[3]	A[11] D'[9]	A[12] D'[15]	A[13] D'[2]	A[14] D'[8]
A[15] D'[14]	A[16] D'[1]	A[17] D'[7]	A[18] D'[13]	A[19] D'[19]

(c)

A[0] D"[0]	A[1] D"[11]	A[2] D"[3]	A[3] D"[14]	A[4] D"[6]
A[5] D"[17]	A[6] D"[9]	A[7] D"[1]	A[8] D"[12]	A[9] D"[4]
A[10] D"[15]	A[11] D"[7]	A[12] D"[18]	A[13] D"[10]	A[14] D"[2]
A[15] D"[13]	A[16] D"[5]	A[17] D"[16]	A[18] D"[8]	A[19] D"[19]

FIG. 6

ADDRESS SET NUMBER x

ADDRESS NUMBER y

	0	1	2	3	4	5	6	7	8	9
0	0	0	0	0	0	0	0	0	0	0
1	1	4	16	7	9	17	11	6	5	1
2	2	8	13	14	18	15	3	12	10	2
3	3	12	10	2	8	13	14	18	15	3
4	4	16	7	9	17	11	6	5	1	4
5	5	1	4	16	7	9	17	11	6	5
6	6	5	1	4	16	7	9	17	11	6
7	7	9	17	11	6	5	1	4	16	7
8	8	13	14	18	15	3	12	10	2	8
9	9	17	11	6	5	1	4	16	7	9
10	10	2	8	13	14	18	15	3	12	10
11	11	6	5	1	4	16	7	9	17	11
12	12	10	2	8	13	14	18	15	3	12
13	13	14	18	15	3	12	10	2	8	13
14	14	18	15	3	12	10	2	8	13	14
15	15	3	12	10	2	8	13	14	18	15
16	16	7	9	17	11	6	5	1	4	16
17	17	11	6	5	1	4	16	7	9	17
18	18	15	3	12	10	2	8	13	14	18
19	19	19	19	19	19	19	19	19	19	19

FIG. 7

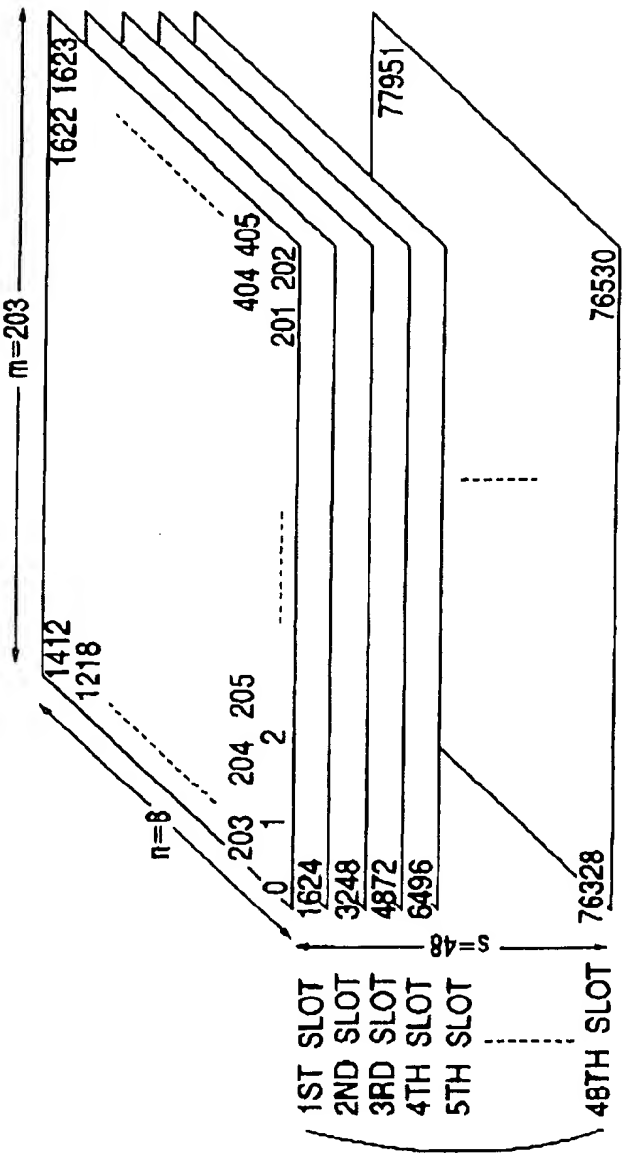

m=203
n=8
0
1
2
203
204
205
1218
1412
201
202
404
405
1622
1623
1624
3248
4872
6496
76328
76530
77951
S=48
1ST SLOT
2ND SLOT
3RD SLOT
4TH SLOT
5TH SLOT
48TH SLOT

FIG. 8

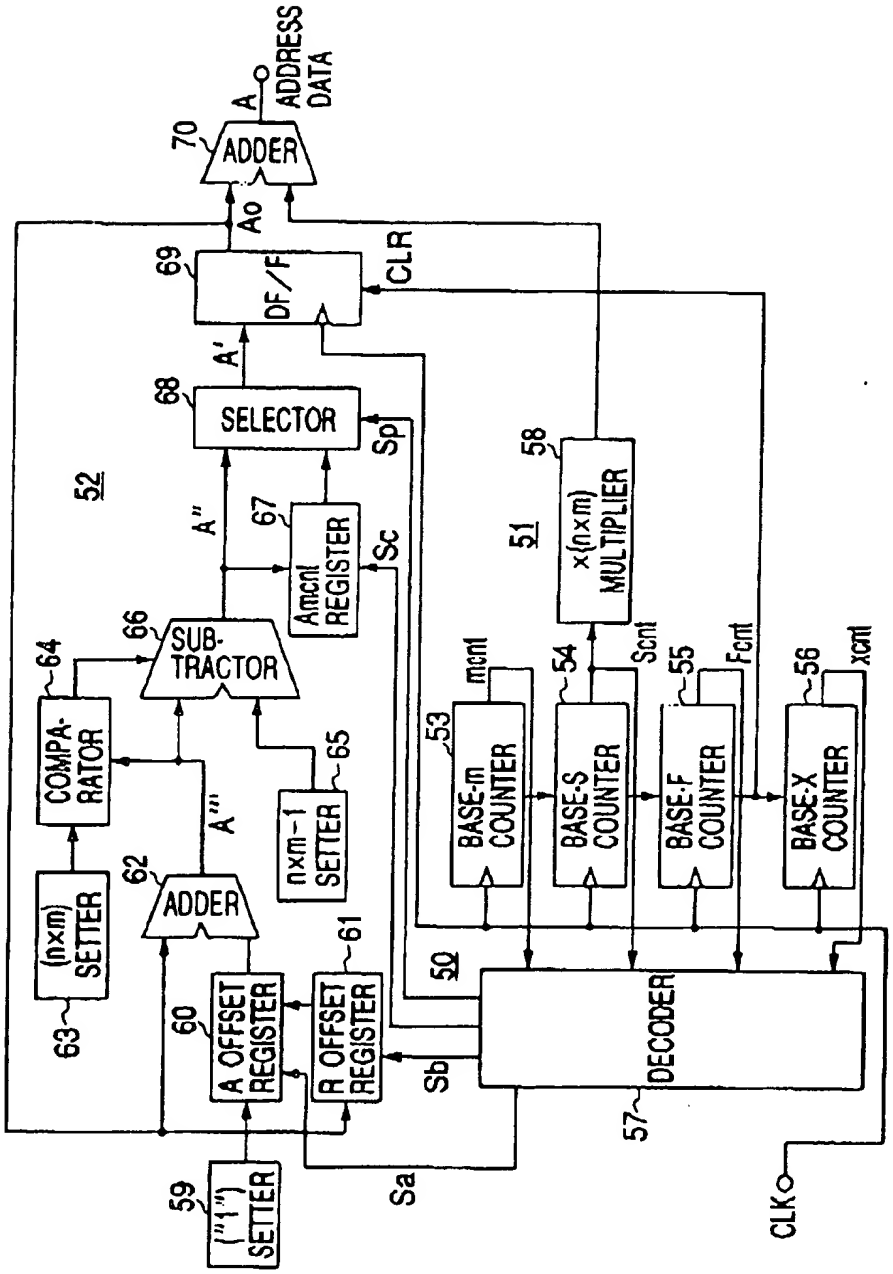

ADDRESS DATA
A
70
ADDER
Ao
69
DF/F
CLR
A'
68
SELECTOR
Sp
52
A"
67
Amcnt REGISTER
Sc
66
SUB-TRACTOR
64
COMPA-RATOR
A'''
62
ADDER
65
n×m-1 SETTER
63
(n×m) SETTER
60
A OFFSET REGISTER
61
R OFFSET REGISTER
59
("1") SETTER
Sa
Sb
50
DECODER
57
CLK
51
58
x(n×m) MULTIPLIER
mcnt
53
BASE-m COUNTER
Scnt
54
BASE-S COUNTER
Fcnt
55
BASE-F COUNTER
xcnt
56
BASE-X COUNTER

FIG. 9

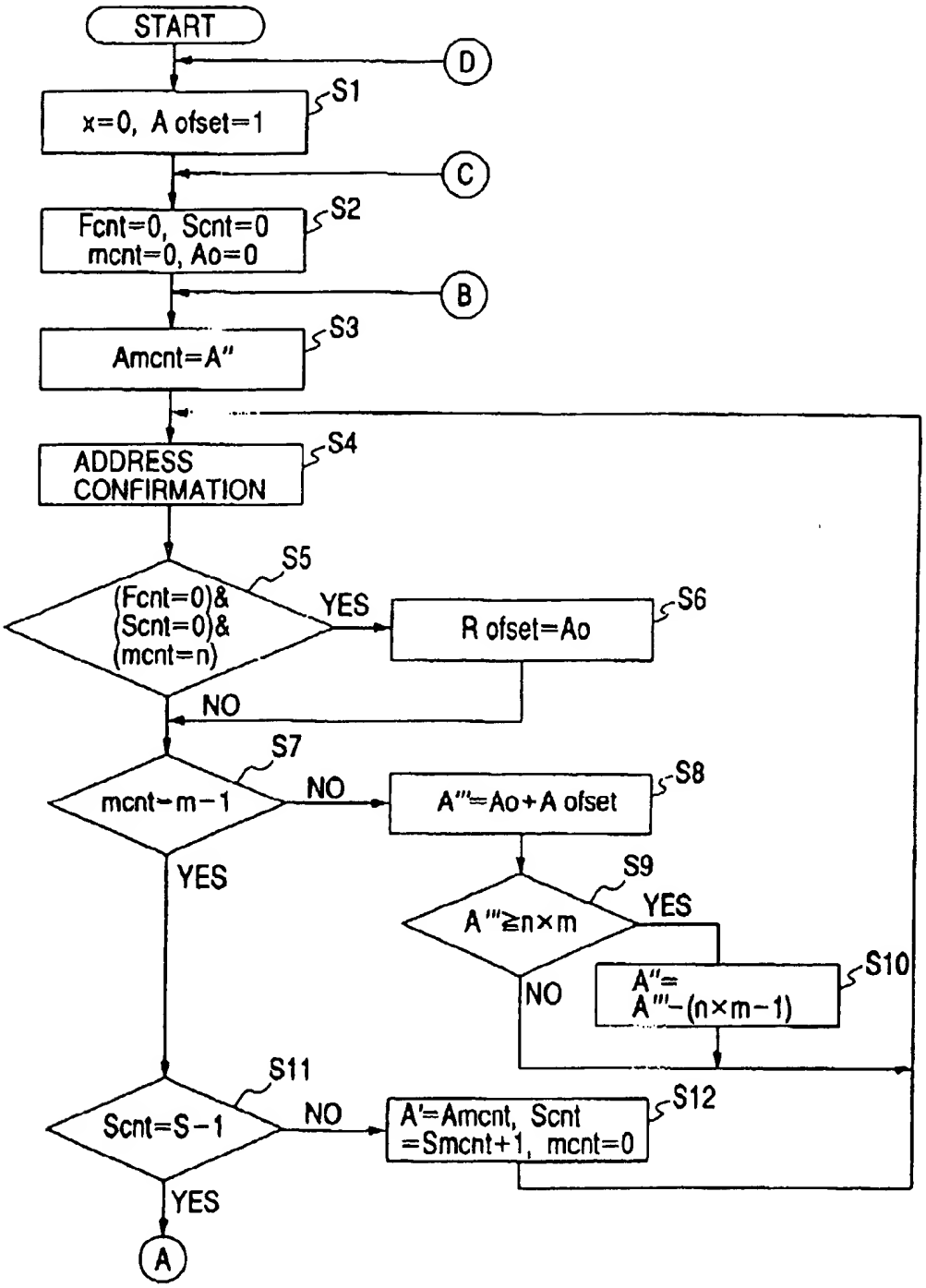

START
D
S1
x=0, A ofset=1
C
S2
Fcnt=0, Scnt=0
mcnt=0, Ao=0
B
S3
Amcnt=A"
S4
ADDRESS CONFIRMATION
S5
(Fcnt=0)&
(Scnt=0)&
(mcnt=n)
YES
S6
R ofset=Ao
NO
S7
mcnt=m-1
NO
S8
A"=Ao+A ofset
YES
S9
A"≧n×m
YES
S10
A"=
A"-(n×m-1)
NO
S11
Scnt=S-1
NO
S12
A'=Amcnt, Scnt
=Smcnt+1, mcnt=0
YES
A

FIG. 10

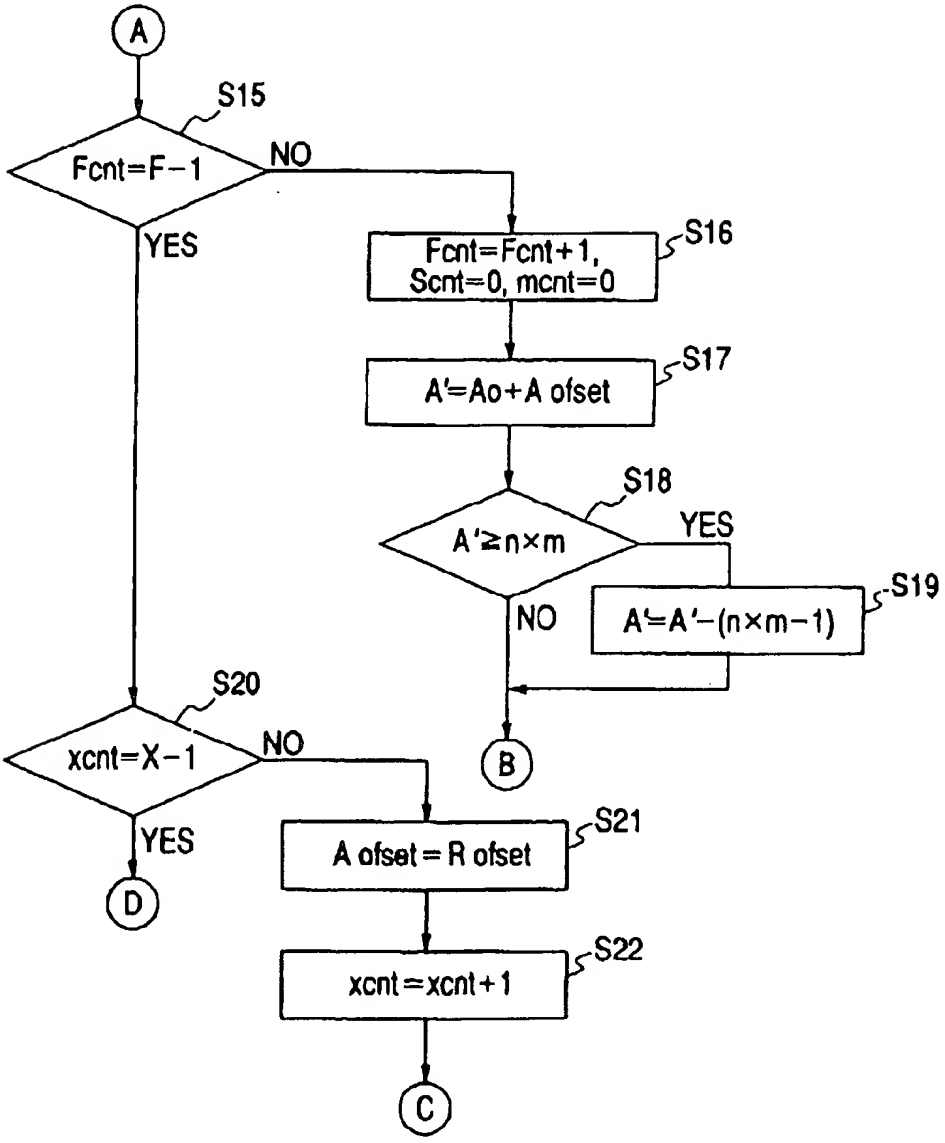

A
S15
Fcnt=F−1
NO
YES
Fcnt=Fcnt+1,
Scnt=0, mcnt=0
S16
A'=Ao+A ofset
S17
S18
A'≧n×m
YES
NO
A'=A'−(n×m−1)
S19
B
S20
xcnt=X−1
NO
YES
D
A ofset = R ofset
S21
xcnt=xcnt+1
S22
C

FIG. 11

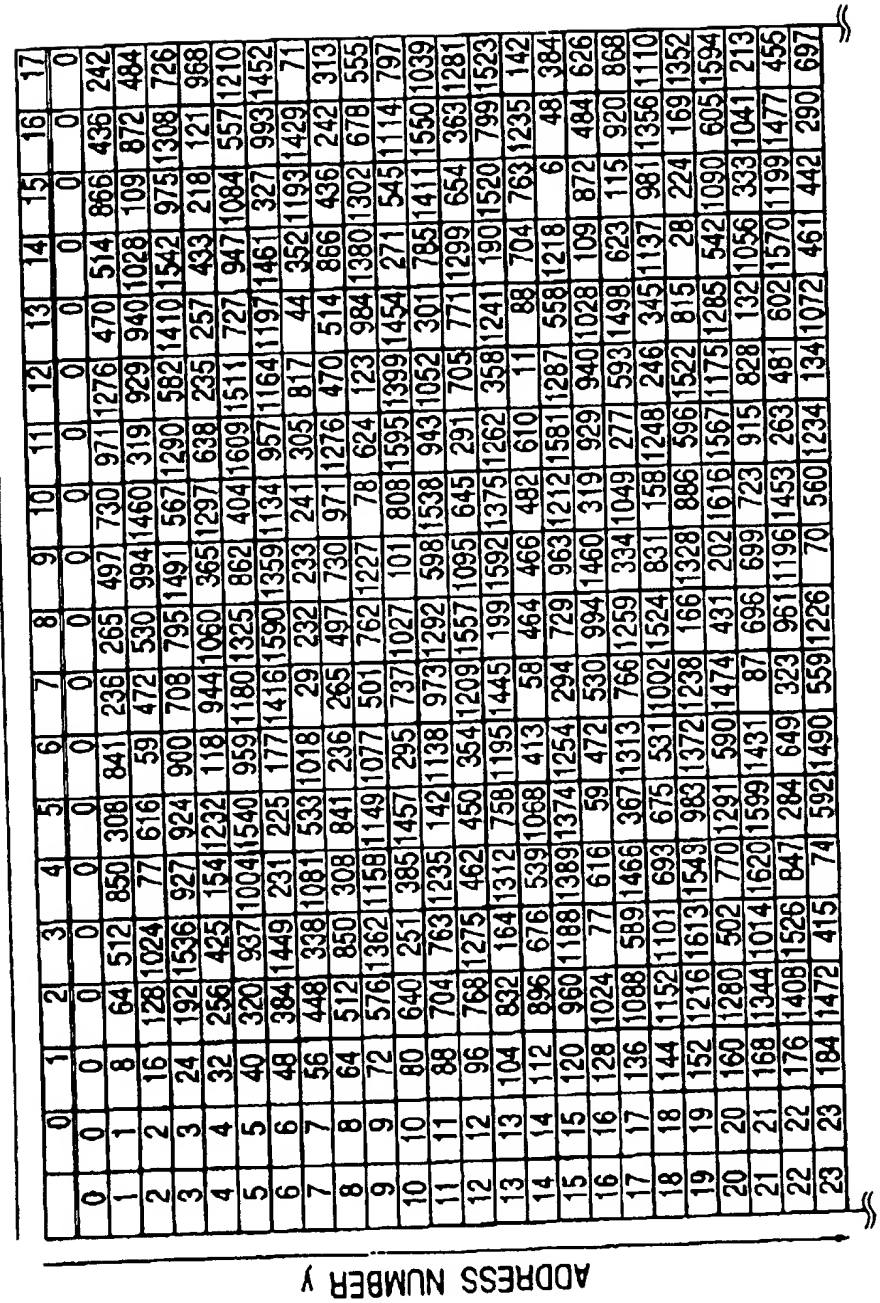
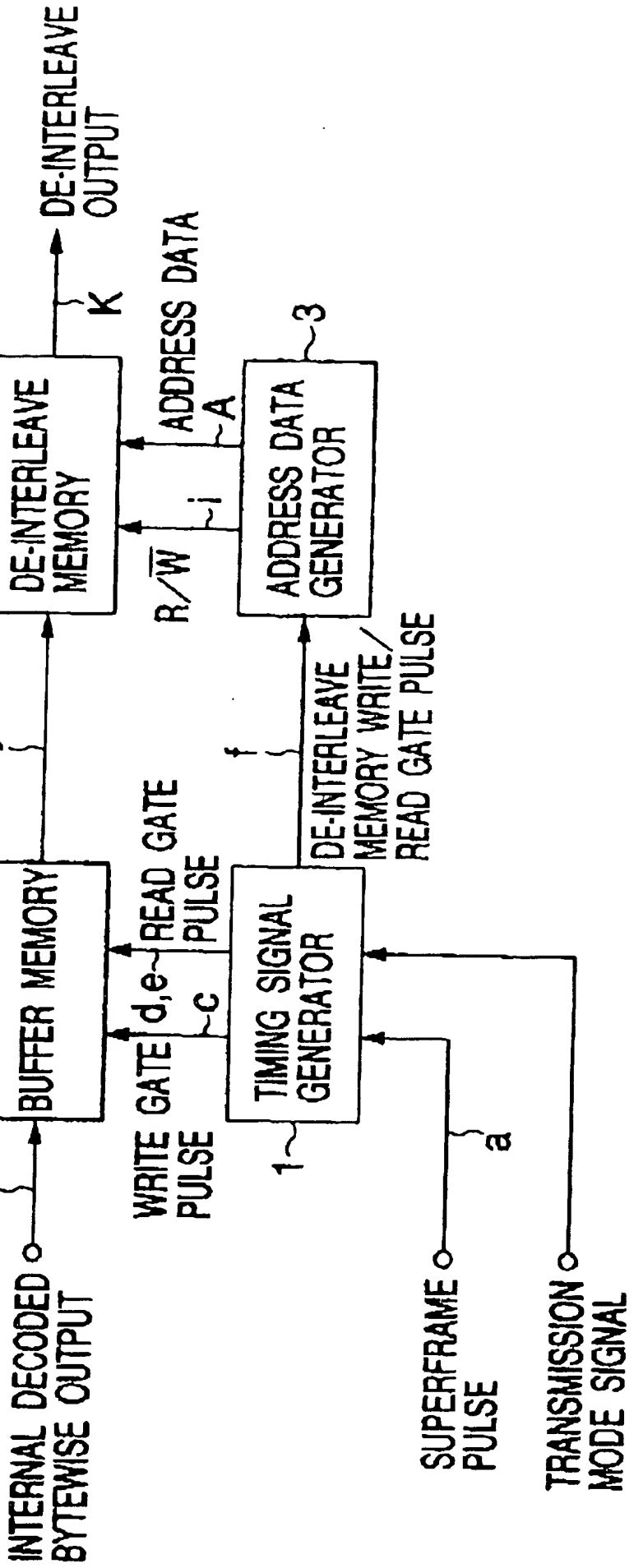


FIG. 12

ADDRESS SET NUMBER x

ADDRESS NUMBER y

	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50
24	24	192	1536	927	924	900	708	795	1491	567	1290	582	1410	1542	975	1308	726	939									
25	25	200	1600	1439	151	1208	1549	1031	133	1064	397	1553	1063	389	1489	551	1162	1181									
26	26	208	41	328	1001	1516	767	1267	398	1561	1127	901	716	859	380	1417	1598	1423									
27	27	216	105	840	228	201	1608	1503	663	435	234	249	369	1329	894	660	411	42									
28	28	224	169	1352	1078	509	826	116	928	932	964	1220	22	176	1408	1526	847	284									
29	29	232	233	241	305	817	44	352	1193	1429	71	568	1298	646	299	769	1283	526									
30	30	240	297	753	1155	1125	885	588	1458	303	801	1539	951	1116	813	12	96	768									
31	31	248	361	1265	382	1433	103	824	100	800	1531	887	604	1586	1327	877	532	1010									
32	32	256	425	154	1232	118	944	1060	385	1297	638	235	257	433	218	121	968	1252									
33	33	264	489	666	459	426	162	1296	630	171	1368	1206	1533	903	732	987	1404	1494									
34	34	272	553	1178	1309	734	1003	1532	895	668	475	554	1186	1373	1246	230	217	113									
35	35	280	617	67	536	1042	221	145	1160	1165	1205	1525	839	220	137	1096	653	355									
36	36	288	681	579	1386	1350	1062	381	1425	39	312	873	492	690	651	339	1089	597									
37	37	296	745	1091	613	35	280	617	67	536	1042	221	145	1160	1165	1205	1525	839									
38	38	304	809	1603	1463	343	1121	853	332	1033	149	1192	1421	7	56	448	338	1081									
39	39	312	873	492	690	651	339	1089	597	1530	879	540	1074	477	570	1314	774	1323									
40	40	320	937	1004	1540	959	1180	1325	862	404	1609	1511	727	947	1084	557	1210	1565									
41	41	328	1001	1516	767	1267	398	1561	1127	901	716	859	380	1417	1598	1423	23	184									
42	42	336	1065	405	1617	1575	1239	174	1392	1398	1446	207	33	264	489	666	459	426									
43	43	344	1129	917	844	260	457	410	34	272	553	1178	1309	734	1003	1532	895	668									
44	44	352	1193	1429	71	568	1298	648	299	769	1283	526	962	1204	1517	775	1331	910									
45	45	360	1257	318	921	876	516	882	564	1266	390	1497	615	51	408	18	144	1152									
46	46	368	1321	830	148	1184	1357	1118	829	140	1120	845	268	521	922	884	580	1394									
47	47	376	1385	1342	998	1492	575	1354	1094	637	227	193	1544	991	1436	127	1016	13									
48	48	384	1449	231	225	177	1418	1590	1359	1134	957	1164	1197	1461	327	993	1452	255									
49	49	392	1513	743	1075	485	634	203	1	8	64	512	850	308	841	236	265	497									
50	50	400	1577	1255	302	793	1475	439	266	505	794	1483	503	778	1355	1102	701	739									


INTERNAL DECODED BYTEWISE OUTPUT
b
2
BUFFER MEMORY
WRITE GATE PULSE
d,e
READ GATE PULSE
c
1
TIMING SIGNAL GENERATOR
a
SUPERFRAME PULSE
TRANSMISSION MODE SIGNAL
j
4
DE-INTERLEAVE MEMORY
R/W
i
ADDRESS DATA
A
3
ADDRESS DATA GENERATOR
f
DE-INTERLEAVE MEMORY WRITE/ READ GATE PULSE
K
DE-INTERLEAVE OUTPUT